## **搭建Nginx服务器**

Apache Nginx Tengine Lighttpd ----主流Nginx Apache Tengine

Tomcat IBM WebSphere Jboss ----与Java连用，主流是Tomcat

看网站用什么开发的，在网站的新标签那按F12，点击网络

创建无交互的nginx用户，目的以nginx身份启动服务，安全

--without-模块 -----禁用功能

[root@proxy ~]# yum -y install gcc pcre-devel openssl-devel //安装依赖包

[root@proxy ~]# useradd -s /sbin/nologin nginx

[root@proxy ~]# tar -xf nginx-1.10.3.tar.gz

[root@proxy ~]# cd nginx-1.10.3

[root@proxy nginx-1.10.3]# ./configure

--prefix=/usr/local/nginx //指定安装路径

--user=nginx //指定用户

--group=nginx //指定组

--with-http\_ssl\_module //开启SSL加密功能

[root@proxy nginx-1.10.3]# make && make install //编译并安装

nginx命令的用法

[root@proxy ~]# /usr/local/nginx/sbin/nginx //启动服务

[root@proxy ~]# /usr/local/nginx/sbin/nginx -s stop //关闭服务

[root@proxy ~]# /usr/local/nginx/sbin/nginx -s reload //重新加载配置文件

[root@proxy ~]# /usr/local/nginx/sbin/nginx –V //查看软件信息

[root@proxy ~]# ln -s /usr/local/nginx/sbin/nginx /sbin/ //软连接,下次直接敲nginx就可以

netstat命令可以查看系统中启动的端口信息，该命令常用选项如下：

-a显示所有端口的信息

-n以数字格式显示端口号

-t显示TCP连接的端口

-u显示UDP连接的端口

-l显示服务正在监听的端口信息，如httpd启动后，会一直监听80端口

-p显示监听端口的服务名称是什么（也就是程序名称） openssl

[root@proxy ~]# netstat -anptu | grep nginx

测试首页文件 [root@client ~]# curl http://192.168.4.5

Nginx Web服务默认首页文档存储目录为/usr/local/nginx/html/，在此目录下默认有一个名为index.html的文件，使用客户端访问测试页面：

http\_ssl\_

**升级Nginx服务器**

编译新版本nginx软

[root@proxy ~]# tar -zxvf nginx-1.12.2.tar.gz

[root@proxy ~]# cd nginx-1.12.2

[root@proxy nginx-1.12.2]# ./configure

--prefix=/usr/local/nginx

--user=nginx http\_ssl\_

--group=nginx

--with-http\_ssl\_module

[root@proxy nginx-1.12.2]# make

注意：不要安装make install ,否则会覆盖

备份老的nginx主程序，并使用编译好的新版本nginx替换老版本

[root@proxy nginx-1.12.2]# mv /usr/local/nginx/sbin/nginx \

>/usr/local/nginx/sbin/nginxold

或者： mv /usr/local/nginx/sbin/nginx{,.bak}

备份旧版本

[root@proxy nginx-1.12.2]# cp objs/nginx /usr/local/nginx/sbin/ //拷贝新版本

[root@proxy nginx-1.12.2]# make upgrade //升级

或者先杀死 killall nginx 再启动 nginx

[root@proxy ~]# /usr/local/nginx/sbin/nginx –v //查看版本

**用户认证**

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf //主配置文件

全局配置（用户名,日志,进程)

worker\_processes 1; //起多少个进程

worker\_connections 1024 并发量

Server {

Listen 80; 端口

Server\_name www.example.com; 域名

root html; 路径

}

注意：内容必须在花括号内，每项内容必须要分号结尾

1.修改Nginx配置文件

auth\_basic "Input Password:"; //认证提示符,随便写

auth\_basic\_user\_file "/usr/local/nginx/pass"; //认证密码文件

(必须加在server花括号内)

2.生成密码文件，创建用户及密码,使用htpasswd命令创建账户文件，需要确保系统中已经安装了httpd-tools

//创建密码文件，注意：密码文件必须和配置文件的一样

[root@proxy ~]# htpasswd -c /usr/local/nginx/pass tom

Password:

//追加用户，不使用-c选项 ,加c覆盖

[root@proxy ~]# htpasswd /usr/local/nginx/pass jerry

**基于域名的虚拟主机**

1.修改Nginx服务配置,添加相关虚拟主机配置如下:

85,94s/#// 把84-94行前面的#去掉，利用其配置

Server\_name 域名；

root 网站根路径;

2.创建网站根目录及对应首页文件

[root@proxy ~]# mkdir /usr/local/nginx/www

[root@proxy ~]# echo "www" > /usr/local/nginx/www/index.html

3.重启nginx服务

[root@proxy ~]# /usr/local/nginx/sbin/nginx -s reload

4.客户端测试

[root@client ~]# vim /etc/hosts

192.168.4.5 www.a.com www.b.com

基于端口的虚拟主机

listen 8080; //端口

基于IP的虚拟主机

listen 192.168.0.1:80; //IP地址与端口

**SSL虚拟主机**

**源码安装Nginx时必须使用--with-http\_ssl\_module参数，启用加密模块，对于需要进行SSL加密处理的站点添加ssl相关指令（设置网站需要的私钥和证书）。**

**加密算法一般分为对称算法、非对称算法、信息摘要。**

**对称算法有：AES、DES，主要应用在单机数据加密。**

**非对称算法有：RSA、DSA，主要应用在网络数据加密。**

**信息摘要：MD5、sha256，主要应用在数据完整性校验、数据秒传等**

1. 生成私钥与证书

注意：在实际企业中生成私钥和证书都不是自己生成的，在自己内部网站可以自己生成

[root@proxy ~]# cd /usr/local/nginx/conf

[root@proxy ~]# openssl genrsa > cert.key //生成私钥

[root@proxy ~]# openssl req -new -x509 -key cert.key > cert.pem

//生成证书

1. 修改Nginx配置文件，设置加密网站的虚拟主机

在最后一个server 配置，只要修改server\_name 域名;

1. 重启nginx服务

[root@proxy ~]# /usr/local/nginx/sbin/nginx -s reload

总结：1.构建nginx 服务

gcc pcre-devel open-devel

创建无交互的用户nginx

tar解包

cd nginx-1.10.3

./configure --user=nginx --group=nginx --with-http\_ssl\_module

make && make install

/usr/local/nginx/sbin/nginx

ln -s /usr/local/nginx/sbin/nginx /sbin

nginx -anputl | grep nginx

2.升级nignx

cd nginx-1.12.2

./configure --user=nginx --group=nginx --with-http\_ssl\_module

make

mv /usr/local/nginx/sbin/nginx{,.bak}

cp objs/nginx /usr/local/nginx/sbin/

killall nginx && nginx

nginx -v

3.用户认证

/usr/local/nginx/conf/nginx/nginx.conf

server{

listen 80;

server\_name localhost;

auth\_basic “dd:”;

auth\_basic\_user\_file “/usr/local/nginx/pass”;

root html;

index index.html index.htm;

}

生成密码文件，创建用户及密码

yum -y install httpd-tools

htpasswd -c /usr/local/nginx/pass tom

password:\*\*

若追加用户

htpasswd /usr/local/nginx/pass harry

password:\*\*

/usr/local/nginx/sbin/nginx -s reload

4.虚拟主机

/usr/local/nginx/conf/nginx.conf

server{

listen 80;

server\_name [www.a.com](http://www.a.com);

auth\_basic “123:”;

auth\_basic\_user\_file “/usr/local/ningx/pass”;

root html;

index index.html index.htm

}

server{

listen 80；

server\_name [www.b.com](http://www.b.com);

location / {

root www;

index index.html index.htm

}

mkdir /usr/local/nginx/www

echo “hello world”> /usr/local/nginx/www/index.html

nginx -s reload

5.SSL虚拟主机

cd /usr/local/nginx/conf

openssl genrsa > cert.key

openssl req -new -x509 -key cert.key > cert.pem 若是对外，私钥和证书就得买

修改配置文件/usr/local/nginx/conf/nginx/nginx.conf

server{

listen 443ssl;

server\_name [www.c.com](http://www.c.com);

......... 打开

nginx -s reload